Week 12 - Neural Network

1. Using the data synthesis R script provided by the instructor as part of the week 11 assignment instructions, produce datasets of the following sizes, and fit deep learning models with the configurations shown below. Associated with each model, record the following performance characteristics: training error, validation (i.e., holdout set) error, time of execution. Use an appropriate activation function.

Data size	Configuration	Training error	Validation error	Time of execution(s)
1000	1 hidden layer 4 nodes	14552467.0555	14419139.8553	9.25 seconds
10000	1 hidden layer 4 nodes	9462526.3514	10213397.9534	32.30 seconds
100000	1 hidden layer 4 nodes	138168947.2204	135043017.2930	319.26 seconds
1000	2 hidden layers of 4 nodes each	73428227.4457	74458843.2386	9.89 seconds
10000	2 hidden layers of 4 nodes each	6405523.1911	6441236.3064	36.11 seconds
100000	2 hidden layers of 4 nodes each	56407593.0875	56622360.8590	349.49 seconds
1000	XGBoost	20576375	23883824	0.01
10000	XGBoost	616872251	638614024	0.01
100000	XGBoost	56935639286	56445635627	0.07

2. Based on the results, which model do you consider as superior, among the deep learning models fit?

The evaluation results for deep learning models indicate clear distinctions depending on both their architecture design along with the number of examples in the dataset. Both training and validation errors indicate that the single hidden layer network exceeds the performance of the two-layer network at 1,000 samples. These results indicate that the two-layer model possibly operates outside its optimal range or it suffers from overfitting issues when working with this limited sample size. Other than 1,000 samples where performance is comparable between the two models the multi-layer architecture outpaces the single-layer architecture when the data sets grow to contain 10,000 samples. The results demonstrate increasing superiority of the two-layer model compared to the single-layer model because its error output declines by approximately half. The execution time spans identically between the two model types which reveals that better performance from two-layer implementation does not entail additional computational expenses.

The two-layer deep learning model outperforms the single-layer model irrespective of dataset size but performs best on large data sets. Deeper networks work better since they scale better as data volume increases and achieve better levels of generalization on big datasets.

3. Next, report the results (for the particular numbers of observations) from applying xgboost (week 11 – provide the relevant results here in a table). Comparing the results from XGBoost and deep learning models fit, which model would you say is superior to others? What is the basis for your judgment?

The implementation of XGBoost demonstrates excellent performance on small datasets while reaching better training and validation errors than deep learning models while running in a shorter period. The deep learning models demonstrate improved accuracy results compared to XGBoost when the training dataset exceeds 10,000 and reaches 100,000 observations particularly when using a two-layer network model. XGBoost maintains speed advantages but its large error rates on bigger datasets show either bad scalability characteristics or indicate the model needs better parameter optimization. The two-layer deep learning model offers superior performance on medium to large datasets because of its higher accuracy although XGBoost demonstrates better speed for small datasets.